

作业提交服务器：ftp://192.168.134.123

用户名：uploader

密码：sdt7%5252@3

！ 注意事项

- 1. 每题需要提交一个cpp文件，确保cpp文件可以正确编译。cpp文件用题目编号命名。
 - 2. 每题需要提交至少一张运行结果截图。将所有题目的截图放在一个pdf文件中。
 - 3. 将所有cpp文件和一个pdf文件打包成zip文件，用自己的学号命名，并上传至ftp指定文件夹。
- 举例说明：比如学号是1001，本次有3个题目，那么最终提交1个1001.zip文件，其中包含3个cpp文件（分别是1.cpp, 2.cpp, 3.cpp）和1个pdf文件，其中pdf文件中包含至少3个运行结果的截图。第1次作业上传至hm1文件夹。

第2次作业

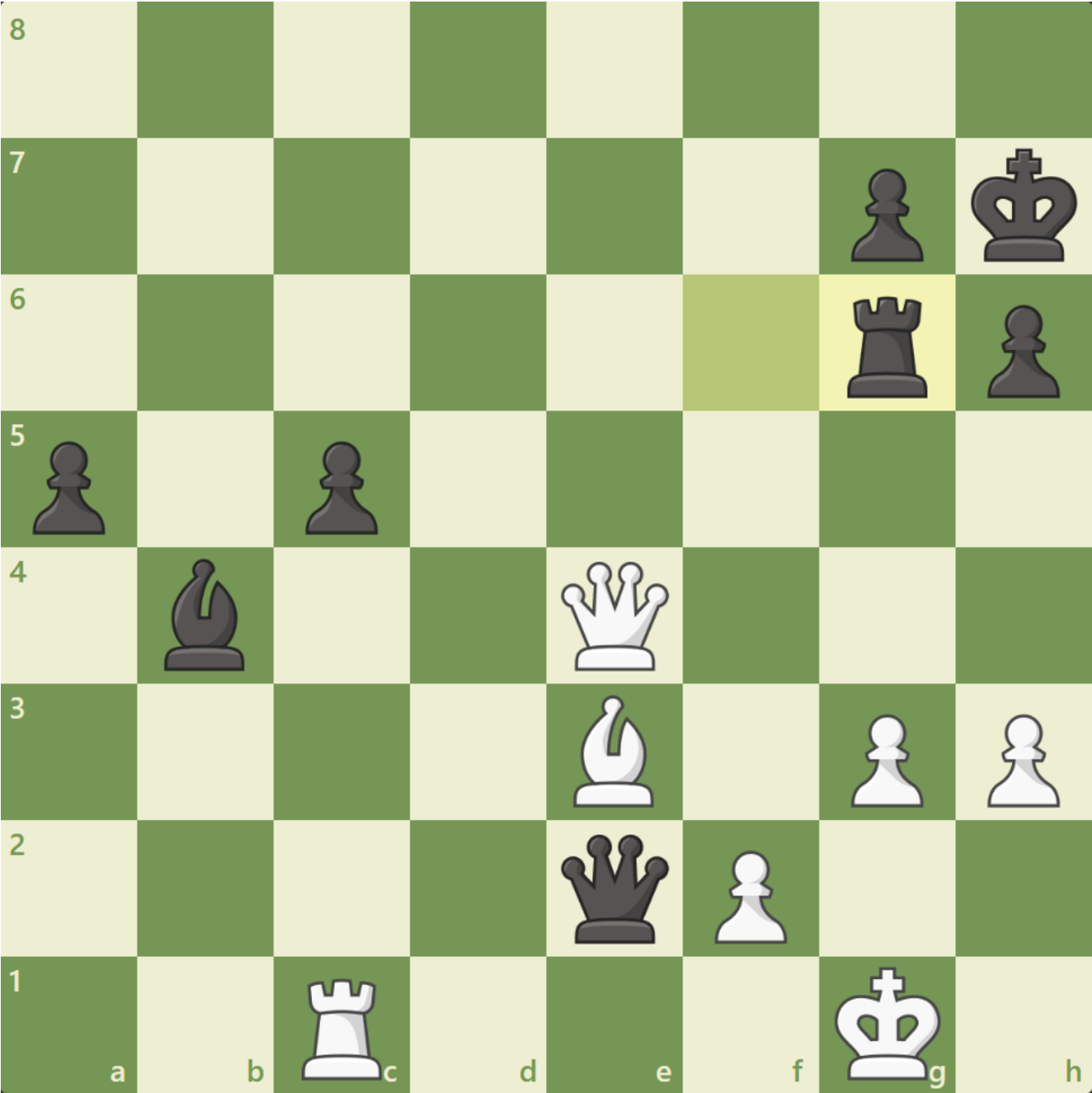
提交时间：2024年4月21日22:00

第一题问题描述

国际象棋在对局时，同一局面连续或间断出现3次或3次以上，可由任意一方提出和棋。

国际象棋每一个局面可以用大小为 8×8 的字符数组来表示，其中每一位对应棋盘上的一个格子。六种棋子王、后、车、象、马、兵分别用字母 **k**、**q**、**r**、**b**、**n**、**p** 表示，其中大写字母对应白方、小写字母对应黑方。棋盘上无棋子处用字符 ***** 表示。两个字符数组的每一位均相同则说明对应同一局面。

现已按上述方式整理好了每步棋后的局面，试统计每个局面分别是第几次出现。



输入格式

从标准输入读入数据。

输入的第一行包含一个正整数 n ，表示这盘棋总共有 n 步。

接下来 $8 \times n$ 行，依次输入第1到第 n 步棋后的局面。具体来说每行包含一个长度为8的字符串，每8行字符串共 64 个字符对应一个局面。

输出格式

输出到标准输出中。

输出共 n 行，每行一个整数，表示该局面是第几次出现。

样例输入

1 8
2 *****
3 *****pk
4 *****r*p
5 p*pQ****
6 *****
7 **b*B*PP
8 *****qP**
9 **R***K*
10 *****
11 *****pk
12 *****r*p
13 p*pQ****
14 *b*****
15 *****B*PP
16 *****qP**
17 **R***K*
18 *****
19 *****pk
20 *****r*p
21 p*p*****
22 *b**Q***
23 *****B*PP
24 *****qP**
25 **R***K*
26 *****k*
27 *****p*
28 *****r*p
29 p*p*****
30 *b**Q***
31 *****B*PP
32 *****qP**
33 **R***K*
34 *****k*
35 *****p*
36 *****r*p
37 p*pQ****
38 *b*****
39 *****B*PP
40 *****qP**
41 **R***K*
42 *****
43 *****pk
44 *****r*p
45 p*pQ****
46 *b*****
47 *****B*PP
48 *****qP**
49 **R***K*
50 *****
51 *****pk
52 *****r*p

```
53 | p*p*****
54 | *b**Q***
55 | ****B*PP
56 | ****qP**
57 | **R***K*
58 | *****
59 | *****pk
60 | *****rp
61 | p*p*****
62 | *b**Q***
63 | ****B*PP
64 | ****qP**
65 | **R***K*
```

样例输出

```
1 | 1
2 | 1
3 | 1
4 | 1
5 | 1
6 | 2
7 | 2
8 | 1
```

样例解释

第6、7 步后的局面分别与第2、3步后的局面相同。第8步后的局面与上图相对应。

子任务

输入数据满足 $n \leq 100$ 。

时间限制：1.0s

内存限制：512.0MB

提示

判断重复局面仅涉及字符串比较，无需考虑国际象棋实际行棋规则。

第二题问题描述

$Softmax(\frac{Q \times K^T}{\sqrt{d}}) \times V$ 是 Transformer 中注意力模块的核心算式，其中 Q 、 K 和 V 均是 n 行 d 列的矩阵， K^T 表示矩阵 K 的转置， \times 表示矩阵乘法。

为了方便计算，顿顿同学将 $Softmax$ 简化为了点乘一个大小为 n 的一维向量 W ： $(W \cdot (Q \times K^T)) \times V$ ，点乘即对应位相乘，记 $W^{(i)}$ 为向量 W 的第 i 个元素，即将第 $(Q \times K^T)$ 第 i 行中的每个元素都与 $W^{(i)}$ 相乘。

现给出矩阵 Q 、 K 和 V 和向量 W ，试计算顿顿按简化的算式计算的结果。

输入格式

从标准输入读入数据。

输入的第一行包含空格分隔的两个正整数 n 和 d ，表示矩阵的大小。

接下来依次输入矩阵 Q 、 K 和 V 。每个矩阵输入 n 行，每行包含空格分隔的 d 个整数，其中第 i 行的第 j 个数对应矩阵的第 i 行、第 j 列。

最后一行输入 n 个整数，表示向量 W 。

输出格式

输出到标准输出中。

输出共 n 行，每行包含空格分隔的 d 个整数，表示计算的结果。

样例输入

```
1 3 2
2 1 2
3 3 4
4 5 6
5 10 10
6 -20 -20
7 30 30
8 6 5
9 4 3
10 2 1
11 4 0 -5
```

样例输出

```
1 480 240
2 0 0
3 -2200 -1100
```

子任务

70% 的测试数据满足： $n \leq 100$ 且 $d \leq 10$ ；输入矩阵、向量中的元素均为整数，且绝对值均不超过30。
全部的测试数据满足： $n \leq 10^4$ 且 $d \leq 20$ 。输入矩阵、向量中的元素均为整数，且绝对值均不超过1000。

时间限制：5.0s

内存限制：512.0MB

提示

请谨慎评估矩阵乘法运算后的数值范围，并使用适当数据类型存储矩阵中的整数。

第三题问题描述

西西艾弗岛上散落着 n 块田地。每块田地可视为平面直角坐标系下的一块矩形区域，由左下角坐标 (x_1, y_1) 和右上角坐标 (x_2, y_2) 唯一确定，且满足 $x_1 < x_2$ 、 $y_1 < y_2$ 。这 n 块田地中，任意两块的交集面积均为0，仅边界处可能有所重叠。

最近，顿顿想要在南山脚下开垦出一块面积为 $a \times b$ 矩形田地，其左下角坐标为 $(0, 0)$ 、右上角坐标为 (a, b) 。试计算顿顿选定区域内已经存在的田地面积。

输入格式

从标准输入读入数据。

输入共 $n + 1$ 行。

输入的第一行包含空格分隔的三个正整数 n 、 a 和 b ，分别表示西西艾弗岛上田地块数和顿顿选定区域的右上角坐标。

接下来 n 行，每行包含空格分隔的四个整数 x_1 、 y_1 、 x_2 和 y_2 ，表示一块田地的位置。

输出格式

输出到标准输出。

输出一个整数，表示顿顿选定区域内的田地面积。

样例输入

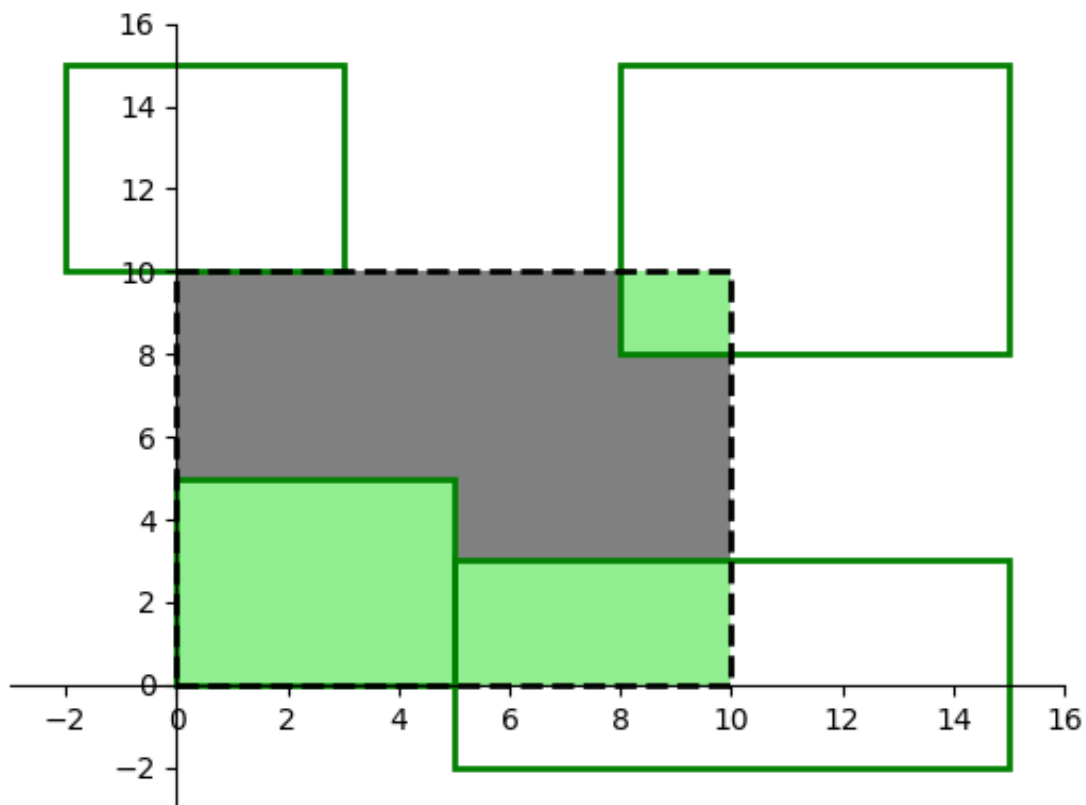
1	4	10	10
2	0	0	5 5
3	5	-2	15 3
4	8	8	15 15
5	-2	10	3 15

样例输出

1	44
---	----

样例说明

如图所示，选定区域内田地（绿色区域）面积为44。



子任务

全部的测试数据满足： $n \leq 100$ ，且所有输入坐标的绝对值不超过 10^4 。

时间限制：1.0s

内存限制：512.0MB

第四题问题描述

顿顿总共选中了 n 块区域准备开垦田地，由于各块区域大小不一，开垦所需时间也不尽相同。据估算，其中第 i 块（ $1 \leq i \leq n$ ）区域的开垦耗时为 t_i 天。这 n 块区域可以同时开垦，所以总耗时 t_{Total} 取决于耗时最长的区域，即： $t_{Total} = \max\{t_1, t_2, \dots, t_n\}$

为了加快开垦进度，顿顿准备在部分区域投入额外资源来缩短开垦时间。具体来说：

- 在第 i 块区域每投入 c_i 单位资源，便可将其开垦耗时缩短1天；
- 耗时缩短天数以整数记，即第 i 块区域投入资源数量必须是 c_i 的整数倍；
- 在第 i 块区域最多可投入 $c_i \times (t_i - k)$ 单位资源，将其开垦耗时缩短为 k 天；
- 这里的 k 表示开垦一块区域的最少天数，满足 $0 < k \leq \min\{t_1, t_2, \dots, t_n\}$ ；换言之，如果无限制地投入资源，所有区域都可以用 k 天完成开垦。

现在顿顿手中共有 m 单位资源可供使用，试计算开垦 n 块区域最少需要多少天？

输入格式

从标准输入读入数据。

输入共 $n + 1$ 行。

输入的第一行包含空格分隔的三个正整数 n 、 m 和 k ，分别表示待开垦的区域总数、顿顿手上的资源数量和每块区域的最少开垦天数。

接下来 n 行，每行包含空格分隔的两个正整数 t_i 和 c_i ，分别表示第 i 块区域开垦耗时和将耗时缩短1天所需资源数量。

输出格式

输出到标准输出。

输出一个整数，表示开垦 n 块区域的最少耗时。

样例输入1

```
1 | 4 9 2
2 | 6 1
3 | 5 1
4 | 6 2
5 | 7 1
```

样例输出1

```
1 | 5
```

样例说明

如下表所示，投入5单位资源即可将总耗时缩短至5天。此时顿顿手中还剩余4单位资源，但无论如何安排，也无法使总耗时进一步缩短。

i	基础耗时 t_i	缩减1天所需资源 c_i	投入资源数量	实际耗时
1	6	1	1	5
2	5	1	0	5
3	6	2	2	5
4	7	1	2	5

样例输入2

```
1 | 4 30 2
2 | 6 1
3 | 5 1
4 | 6 2
5 | 7 1
```

样例输出2

样例说明

投入20单位资源，恰好可将所有区域开垦耗时均缩短为 $k = 2$ 天；受限于 k ，剩余的10单位资源无法使耗时进一步缩短。

子任务

70%的测试数据满足： $0 < n, t_i, c_i \leq 100$ 且 $0 < m \leq 10^6$ ；

全部的测试数据满足： $0 < n, t_i, c_i \leq 10^5$ 且 $0 < m \leq 10^9$ 。

时间限制：1.0s

内存限制：512.0MB